Stack Navigation

En este commit, lo que se desea conseguir es crear un menú de navegación entre el componente *Calendario* (que será el primero en cargarse cuando se lance la aplicación) y el componente *DetalleExcursión* que se renderiza al hacer click en uno de los *ListItems* de *Calendario*.

1. Stack Navigation

Inicialmente se instala en la carpeta principal en la que se esta trabajando la librería React Navigation, junto con sus correspondientes dependencias:

yarn add @react-navigation/native
expo install react-native-gesture-handler
expo install react-native-reanimated
expo install react-native-screens
expo install react-native-safe-area-context
expo install @react-native-community/masked-view

Además, ahora instalamos React Navigation Stack:

yarn add @react-navigation/stack

Ahora ya se puede proceder a realizar los cambios apropiados para conseguir nuestro objetivo. El fichero CampoBaseComponent.js será en el que se defina el menú de navegación, para ello se realizan los siguientes cambios:

- En el componente CampoBase se importa el NavigationContainer de la librería @react-navigation/native. El NavigationContainer es el responsable de administrar el estado de la aplicación y vincular el navegador de nivel superior al entorno de la aplicación. El contenedor proporciona varias funciones útiles:
  - Integración de enlace gracias a linking prop.
  - Notifica los cambios de estado para el seguimiento de la pantalla, la persistencia del estado, etc.
  - Maneja el botón de retroceso del sistema en Android usando la BackHandlerAPI de React Native.

- También en este mismo componente se importa createStackNavigation de la librería @react-navigation/stack. La función createStackNavigation se iguala a la constante Stack, la cual se utiliza para Stack.Navigator y Stack.Screen. Esta función proporciona una forma para que la aplicación pase de una pantalla a otra, donde cada nueva pantalla se coloca en la parte superior de un stack.
  Por tanto, dentro del Stack.Navigator se tendrán todas las pantallas que se desea que contenga el navegador. Dentro de las Stack.Screen se define una de las pantallas dentro del navegador.

- El componente *Calendario* envía un parámetro a *DetalleExcursion*. Gracias a:

```
const { navigate } = this.props.navigation;
```

  Se indica el nombre de la ruta: DetalleExcursion y el parámetro que se desea pasar.

```
onPress={() => navigate('DetalleExcursion', { excursionId: item.id })}
```

En el componente DetalleExcursion se obtiene este parámetro utilizando:

```
const {excursionId} = this.props.route.params;
```

Finalmente, se consigue lo deseado: